

Documentation Templates and Rules (DTR) API User Guide



Prepared By:
Acentra Health
1600 Tysons Boulevard, Suite 1000
McLean, VA 22102



evoBrix X v2026-06
July 08, 2026

Table of Contents

Preface.....	1
Purpose and Scope.....	1
Document Conventions	1
Audience Requirements	1
Chapter 1 – Overview	2
1.1 Data Flow	3
Chapter 2 – Documentation Templates and Rules (DTR) API.....	4
2.1 Overview	4
2.1.1 Components of Documentation Templates and Rules (DTR) API.....	5
2.1.2 DTR Launch and Processing of DTR Request.....	5
2.1.3 DTR Response Process	6
2.1.4 DTR API Documentation	6
3.2 FHIR Server Sandbox Environment.....	8
3.3 Accessing the Sandbox Environment	8
3.4 Base URLs	8
3.5 Capability Statement	9
3.6 Supported Scopes.....	9
3.7 Resources	9
3.8 Request Headers	10
Chapter 3 – Connecting to Documentation Templates and Rules (DTR) API.....	11
Chapter 4 – Third-Party App Registration	12
4.1 Acentra Sandbox Environment	12
4.2 Environments	12
Chapter 5 – Glossary	13
Chapter 6 – Acronyms	14

Tables

Table 1: Documentation Templates and Rules (DTR) API Solution Components.....	2
Table 2: Online Documentation - DTR API.....	6
Table 3: Base URLs for Base URLs for DTR API.....	9
Table 4: Resources and URLs for and URLs for DTR API	10
Table 5: Request Headers	10
Table 6: Glossary.....	13
Table 7: Acronyms.....	14

Figures

Figure 1 Acentra Interoperability Solution- DTR API	2
Figure 2: Acentra Sandbox Environment – Online Developer Documentation.....	8
Figure 3: Third-Party App Registration Process	12

Preface

Purpose and Scope

The DTR API allows EHR/EMR systems to initiate Request to the registered DTR Payer Service that enables Payers to express their documentation requirements in a computable format using FHIR Questionnaires. The DTR Payer Service then returns a FHIR-complaint Response to the DTR Clients, allowing EHR/EMR systems to easily gather and submit necessary information.

This document describes Acentra's Interoperability Solution that supports the new CMS mandate.

Document Conventions

This document uses the following conventions:

- The term "you" refers to the application owners who use this document.
- Hyperlink names appear in bold fonts.

Audience Requirements

This document is intended for third-party application owners or developers who need to know how to connect their applications to Acentra's Documentation Templates and Rules (DTR) API, and for Payers implementing the DTR API to provide data to third-party applications via the API.

Chapter 1 – Overview

The Acentra Interoperability Solution is engineered to support the business and technical requirements defined in the CMS mandate (CMS Final Rule 0057 F). The solution leverages **Da Vinci - Documentation Templates and Rules 2.2.0** - STU 2.2 as anchors in conjunction with data exchange standards, such as HL7® Fast Healthcare Interoperability Resources (FHIR®), United States Core Data and Interoperability (USCDI), and CMS recommended Implementation Guides (IGs).

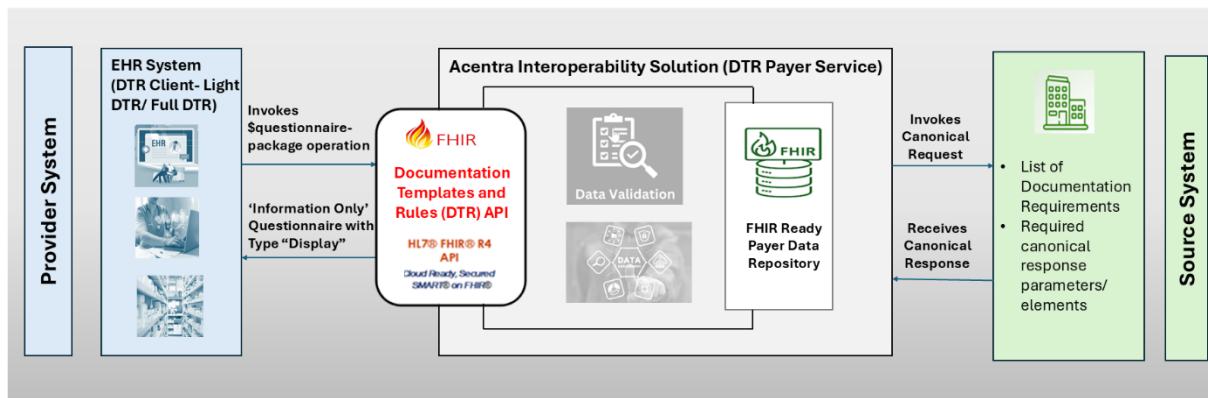


Figure 1 Acentra Interoperability Solution- DTR API

The table below describes the features of the components of the Acentra Interoperability Solution.

Table 1: Documentation Templates and Rules (DTR) API Solution Components

Component	Features
FHIR Server	Implements a RESTful FHIR API service, that supports applicable profiles and related frameworks based on Da Vinci - Documentation Templates and Rules Implementation Guide and aligned with CMS-0057-F Final Rule. Performs four levels of validation on the request payload received from the DTR Client. This includes JSON structure validation, mandatory elements and conditionally required elements validation, validating that the coded elements adhere to the allowed code sets defined in the Implementation Guide (IG). Once these validations are complete, the FHIR Server generates a canonical request structure and invokes internal source system. The FHIR Server, then receives the canonical response from Source System, prepares a FHIR Compliant Response enabling seamless interoperability across Systems. Enables transformation of request payload from FHIR Server to Canonical request structure and Canonical response to FHIR compliant Response and responds back to DTR Client.

Component	Features
Data Repository	All request data elements and response data elements are stored in JSON format on the Audit table in FPDR for tracking and auditing purposes. FHIR Ready Payer Data Repository (FPDR) enables inter-related models for storing and retrieving data via logical views for generating Responses on documentation requirements.
FHIR Adapter	Facilitates the retrieval of data from FPDR and act as integration layer between the FHIR Server and FPDR.
SMART on FHIR Module	Secures the Documentation Templates and Rules (DTR) API by implementing the CMS-recommended standalone launch authorization sequence based on the OAuth 2.0 standard. DTR Client (EHR/EMR Systems) connecting to the FHIR server must authenticate through a manual process over secure TLS channels. The DTR API integrates with SMART® on FHIR to obtain an access token via OAuth 2.0 authorization. The registered and authorized DTR Client (EHR System) is permitted to invoke payer's \$questionnaire-package operation to determine documentation requirements.

1.1 Data Flow

The DTR client initiates a Request to the registered DTR Payer Service and invokes the Payer's **\$questionnaire-package operation**. This occurs either through a Standalone DTR launch without a prior CRD trigger or through a DTR launch initiated after a CRD response. The FHIR Server performs four levels of validation on the request payload received from the DTR client. The FHIR server, based on the validated request data, generates a canonical request structure and invokes internal Source System web services to determine documentation requirements. The FHIR Server, then receives the canonical response from Source System, constructs a FHIR-compliant response and returns an "Information Only" Questionnaire to the DTR Clients, enabling Providers to easily gather and submit the required documentation.

Chapter 2 – Documentation Templates and Rules (DTR) API

2.1 Overview

Documentation Templates and Rules (DTR) API facilitates the secure movement of documentation requirement information data from Payer Systems (Producers) to authorized Providers (Consumers) accessible via authorized third-party consumer applications. It leverages the CMS-recommended Da Vinci - Documentation Templates and Rules) Implementation Guide (IG) in conjunction with data exchange standards, such as **HL7 FHIR**, United States Core Data and Interoperability (**USCDI**), and **SMART on FHIR** Application Launch Framework (Standalone Sequence).

Acentra's Interoperability Solution ensures that the data in transit and at rest is encrypted. To ensure authorized access, EHR System connecting with the FHIR server (DTR Payer Service) must authenticate through a manual process over secure TLS channels. The DTR API integrates with SMART® on FHIR to obtain an access token via OAuth 2.0 authorization. The registered and authorized DTR client will be allowed to invoke \$questionnaire-package operation to determine documentation requirements.

2.1.1 Components of Documentation Templates and Rules (DTR) API

DTR Client:

A DTR Client (EHR/EMR System) can be either a **Light DTR** (EHR-integrated SMART application) or **Full DTR** (native EHR app) that initiates Requests to the registered DTR Payer Service, triggered either through a Standalone DTR launch without a prior CRD trigger.

DTR Payer Service:

DTR Payer Service is the FHIR System that generates canonical structure and invokes internal Payer Systems to retrieve documentation requirements information. Upon receiving canonical responses from the source Systems, the DTR Payer service maps and transforms the data into **FHIR compliant Response** and returns them to the DTR Clients.

2.1.2 DTR Launch and Processing of DTR Request

DTR Launch Approaches:

The DTR client initiates a Request to the registered DTR Payer Service and invokes the Payer's **\$questionnaire-package operation**. This occurs either through a **Standalone DTR launch** without a prior CRD trigger or through a **DTR launch after CRD** response that indicates documentation requirements.

Processing of DTR Request:

- The DTR Client invokes the \$questionnaire-package operation. The “**DTR Questionnaire Package Input Parameters**” profile is used to define the inputs of the \$questionnaire-package operation.
- Once the DTR Payer Service receives a request from the DTR Client, the input payload is validated across four levels:
 - **Level 1** – Syntax Integrity
 - **Level 2** – FHIR Structural Requirements
 - **Level 3** – Inter-Segment (Relational) Validation
 - **Level 4** – Code Sets validation
- The DTR Payer Service validates the request received from the DTR Client by performing the **JSON Structure Validation** to ensure compliance with the IG specification. The **Element validations** will be performed by validating the presence of Coverage Profile elements, Referenced CRD Request profile elements, conditionally required elements based on the specific root or parent element and inter segment relational elements in input parameter profiles. Also, validation of the **Coded elements**

with required conformance that adhere to the allowed code sets defined in the Implementation Guide

- If the validations are unsuccessful, the System must return an Operation Outcome response with relevant diagnostic messages and halt further processing.
- Upon successful validation, The FHIR System generates a canonical request structure by mapping the FHIR parameters into their corresponding Canonical parameters and then invokes the **Internal Source System** with the Canonical request payload to initiate the determination process for the list of documentation requirements.

2.1.3 DTR Response Process

Once a canonical response is received from the **Internal Source System** (e.g., evoBrix X), the DTR Payer Service extracts the available data elements from the Canonical Response and maps them to a FHIR-Compliant Response constructed by the DTR Payer Service. The Response should be returned to the DTR Client which helps in determining documentation requirements.

DTR Questionnaire Package Bundle profile:

- If internal Source Subsystem service has returned a list of documents identified, then it must be sent as part of the Package Bundle parameter in the resource “**DTR Questionnaire Package Bundle**”. The Response must be a Parameters resource conformant to the “**DTR Questionnaire Package Output Parameters**” profile.
- The “**entry:questionnaire**” slice will be used to send the list of documents as an ‘Information only’ questionnaire though R4 Questionnaire resource.
- The “**entry:questionnaireresponse**” slice will be used to pre-populate “Subject”, “Coverage” and other details that FHIR System has, based on the information already present in the FPDR.

Audit Logging must be done for traceability purposes that include the details of validated request/invalid request/ response constructed and returned.

2.1.4 DTR API Documentation

The table below provides links to the online documentation for the DTR API.

Table 2: Online Documentation - DTR API

Function	Online Documentation
IG References	DTR API IG Documentation Templates and Rules Implementation Guide Home Page - Da Vinci - Documentation Templates and Rules v2.2.0

Function	Online Documentation
DTR Request profiles	DTR Questionnaire Package Input Parameters profile DTR Questionnaire Package Input Parameters - Da Vinci - Documentation Templates and Rules v2.2.0 Questionnaire Package Operation \$questionnaire-package CRD Coverage CRD Coverage - Da Vinci - Coverage Requirements Discovery v2.2.0
DTR Response profiles	DTR Questionnaire Package Output Parameters profile DTR Questionnaire Package Output Parameters - Da Vinci - Documentation Templates and Rules v2.2.0-ballot DTR Questionnaire Package Bundle DTR Questionnaire Package Bundle - Da Vinci - Documentation Templates and Rules v2.2.0 R4 Questionnaire Resource Questionnaire - FHIR v4.0.1 DTR Questionnaire Response DTR Questionnaire Response - Da Vinci - Documentation Templates and Rules v2.2.0
Error Codes	 FHIR-Error%20Codes _DTR%20API.xlsx

3.2 FHIR Server Sandbox Environment

Acentra's Interoperability Solution's Sandbox Environment provides a synthetic but reliable dataset of applicable profiles in DTR API to help facilitate testing of the DTR API. The Sandbox dataset represents real data and ensures consistency across the dataset.

Detailed information about Acentra's Interoperability Solution Sandbox Environment can be found at <https://sandbox.mhbapp.com/index.html>. The site includes information for contacting Acentra to register your application.

As shown in Figure 3, the Acentra Sandbox FHIR server documentation portal provides all the required information for the third-party app developers to register their application and connect and access the Sandbox FHIR server for DTR API.

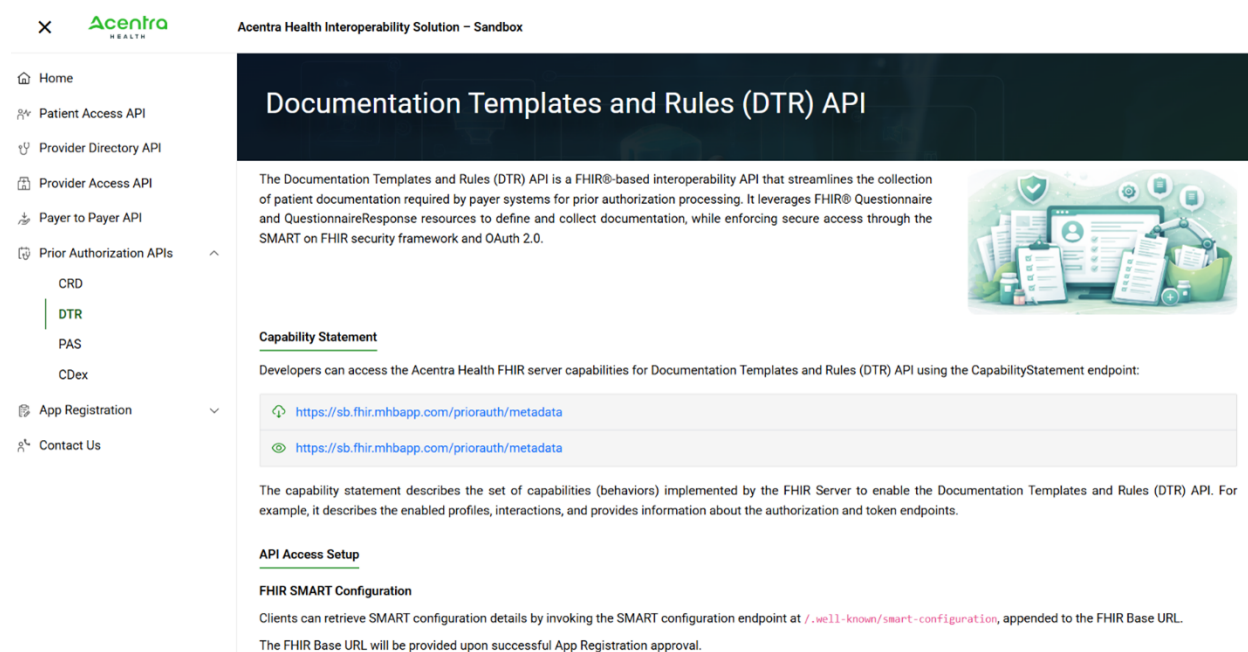


Figure 2: Acentra Sandbox Environment – Online Developer Documentation

3.3 Accessing the Sandbox Environment

Before you can access the DTR API, you will need to register your application with Acentra's Sandbox Environment. Please refer to the Sandbox FHIR server documentation information at <https://sandbox.mhbapp.com/index.html>.

3.4 Base URLs

Three types of FHIR server URLs are provided below for DTR API.

- Acentra Sandbox
- UAT

- Production

The table below provides the base URLs for the FHIR server for the DTR API. Please note that the base URLs are not accessible via browser and are meant to be accessed programmatically.

Table 3: Base URLs for Base URLs for DTR API

Environment	Base URL DTR API
Acentra Sandbox	https://sb.fhir.mhbapp.com/priorauth/api/v1/
UAT	<a href="https://<<tenant-id>>.np.fhir.mhbapp.com/priorauth/api/v1/">https://<<tenant-id>>.np.fhir.mhbapp.com/priorauth/api/v1/
Production	<a href="https://<<tenant-id>>.fhir.mhbapp.com/priorauth/api/v1/">https://<<tenant-id>>.fhir.mhbapp.com/priorauth/api/v1/

3.5 Capability Statement

The capability statement describes the set of capabilities (behaviors) implemented by the FHIR Server; for example, it describes the enabled profiles, interactions, and provides information about the authorization and token endpoints.

The FHIR server capabilities for DTR API can be accessed using the Capability Statement endpoint: https://<<base_url_priorauth>>/metadata.

Please refer to Table 3 for the base URLs for DTR API. For example - the sandbox environment Capability URL will be: <https://sb.fhir.mhbapp.com/priorauth/api/v1/metadata>.

3.6 Supported Scopes

The FHIR Authorization Server supports the following scopes to enable secure, backend system-to-system access for DTR API.

- system/Appointment.read
- system/CommunicationRequest.read
- system/Coverage.read
- system/DeviceRequest.read
- system/Encounter.read
- system/MedicationRequest.read
- system/NutritionOrder.read
- system/ServiceRequest.read
- system/VisionPrescription.read
- system/Questionnaire.read
- system/QuestionnaireResponse.read
- system/OperationOutcome.read

3.7 Resources

The Acentra Interoperability Solution's Documentation Templates and Rules (DTR) API supports the resources listed in Table 4.

Table 4: Resources and URLs for and URLs for DTR API

Resources	URL
Questionnaire	https://{{base_url}}/Questionnaire/\$questionnaire-package

3.8 Request Headers

All requests for Prior Authorization (DTR) API resources require the headers outlined in Table 5.

Table 5: Request Headers

Headers	Values
content-type	application/json
accept	application/json
authorization	Bearer <<auth_code>>

Chapter 3 – Connecting to Documentation Templates and Rules (DTR) API

- The DTR Client must register with the DTR Payer Service through a manual process over TLS to ensure authorized access.
- Initial setup of connectivity between client and Payer will have a manual component to establish security credentials and a trust relationship.
- The DTR Client must obtain an access token via OAuth 2.0 authorization using **SMART on FHIR**.
- Registration Process will ensure the following:
 - Knows the relevant Payer endpoint to access the **\$questionnaire-package operation**.
 - Possess a secret code allowing secure access to the payer endpoint via SMART on FHIR Backend Services.
- After successful registration and authentication, the DTR Client can invoke the \$questionnaire-package operation to retrieve documentation requirements from the DTR Payer Service.

Chapter 4 – Third-Party App Registration

4.1 Acentra Sandbox Environment

The Acentra Sandbox Environment is available for third-party app owners to perform connectivity and testing. Third-party app owners can approach Acentra directly or via payer organization to register their application. If the payer organization approves the request, then Acentra will provide access. In cases where the third-party app owner approaches Acentra directly, it will be Acentra's discretion to approve or deny the request.

4.2 Environments

In the Acentra Interoperability Solution, the intake process collects the registration and attestation information from the third-party app owner and sends it to the authorized personnel within the payer organization (or Tenant). The payer organization reviews the information provided by Acentra to conduct third-party app verification.

If approved, the payer organization notifies Acentra, which then issues credentials (client-id/secret) to third-party application owners for the appropriate environment: Production. If the application registration is not approved, then the payer organization notifies the application owner directly and conveys the decision to Acentra of the outcome. A graphical representation of this process is illustrated in Figure 2.

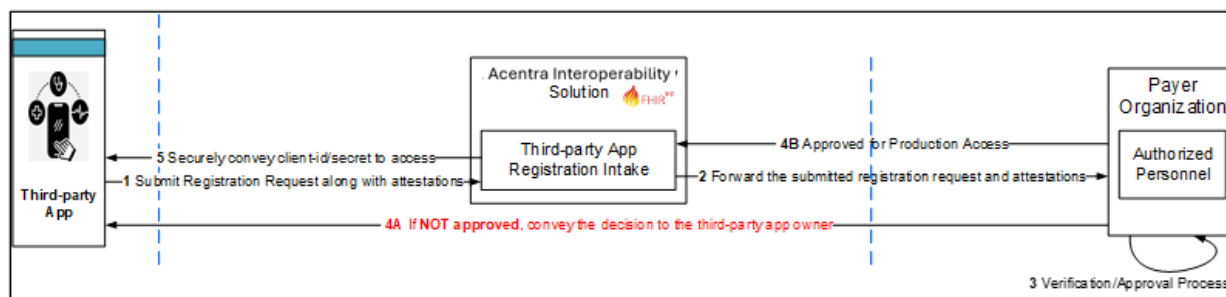


Figure 3: Third-Party App Registration Process

Chapter 5 – Glossary

This section provides definitions for document-specific terms in the following format: Entries are in alphabetical order, and items beginning with numbers will come first.

Table 6: Glossary

Term	Definition
Application Programming Interface	An interface that defines interactions between multiple software applications.
Acentra Interoperability Solution	A solution created to support the business and technical requirements defined in the CMS mandate, CMS Final Rule 0057-F.
Payer Organization	A business entity that adjudicates and renders payment for health care claims, including pharmacy claims. Examples include Medicare, state Medicaid programs, and commercial insurance carriers. In this document, "Tenant" is also used to indicate the Payer Organization in relation to the environment.
Sandbox	An environment containing hardware, instrumentation, simulators, software tools, and other support. Also known as a "test bed".
Tenant	See "Payer Organization".
Third-Party App	An application or program that is neither Acentra nor the Washington State Health Care Authority, that has authorized access via the API for the purpose of delivering data to authorized users (Providers).

Chapter 6 – Acronyms

This section provides definitions for document-specific terms in the following format: Entries are listed in alphabetical order, and items beginning with numbers will come first.

Table 7: Acronyms

Acronym	Definition
API	Application Programming Interface
CMS	Centers for Medicare & Medicaid Services
CDS	Clinical Decision Support
CRD	Coverage Requirement Discovery
DTR	Documentation Templates and Rules
EHR	Electronic Health record
FHIR	Fast Healthcare Interoperability Resources
FPDR	FHIR Ready Payer Data Repository
ID	Identification
IG	Implementation Guide
JSON	JavaScript Object Notation
JWT	JSON Web Token
HIPAA	Health Insurance Portability and Accountability Act
HTTP	HyperText Transfer Protocol
TLS	Transport Layer Security
REST	Representational State Transfer
SMART	Substitutable Medical Applications and Reusable Technologies
STU	Standard for Trial Use
URL	Uniform Resource Locator
USCDI	United States Core Data and Interoperability